**Publisher.py:**

from google.cloud import pubsub\_v1

import json

project\_id = 'dataeng-activity'

topic\_name = 'MyTopic'

publisher = pubsub\_v1.PublisherClient()

topic\_path = publisher.topic\_path(project\_id, topic\_name)

# Function to publish messages from JSON file

def publish\_messages\_from\_file(filename):

with open(filename, 'r') as f:

data = json.load(f)

for record in data:

message\_data = json.dumps(record).encode('utf-8')

future = publisher.publish(topic\_path, data=message\_data)

print(f"Published message: {future.result()}")

# Publish messages from each JSON file

publish\_messages\_from\_file('3951\_vehicle\_data.json')

publish\_messages\_from\_file('3235\_vehicle\_data.json')

**Subscriber.py:**

from google.cloud import pubsub\_v1

project\_id = "dataeng-activity"

subscription\_id = "MySub"

def receive\_messages(project\_id, subscription\_id):

subscriber = pubsub\_v1.SubscriberClient()

subscription\_path = subscriber.subscription\_path(project\_id, subscription\_id)

def callback(message):

print(f"Received message: {message.data.decode('utf-8')}")

message.ack()

streaming\_pull\_future = subscriber.subscribe(subscription\_path, callback=callback)

print(f"Listening for messages on {subscription\_path}..")

# Keep the receiver script running to continuously consume messages

try:

streaming\_pull\_future.result() # Blocking call

except Exception as e:

print(f"Error occurred: {e}")

if \_\_name\_\_ == "\_\_main\_\_":

receive\_messages(project\_id, subscription\_id)

1. What happens if you run your receiver multiple times while only running the publisher once?

Ans:

If you run your receiver multiple times while only running the publisher once, the receiver will consume messages from the Pub/Sub topic each time it runs. However, since there is only one instance of the publisher producing messages, the same set of messages will be consumed each time by the receiver. This means that the receiver will process the same messages repeatedly, potentially leading to duplicated processing.

2. Before the consumer runs, where might the data go, where might it be stored?

Ans:

Before the consumer runs, the data sent by the publisher might be stored temporarily within Pub/Sub. Pub/Sub is a fully managed messaging service provided by Google Cloud, designed to handle large-scale, real-time message ingestion and delivery. When a publisher sends a message to a Pub/Sub topic, the message is stored temporarily in the topic until it is acknowledged by a subscriber. If no subscriber is actively consuming messages from the topic, the messages will remain in the topic for a configurable retention period.

3. Is there a way to determine how much data PubSub is storing for your topic? Do the PubSub monitoring tools help with this?

Ans:

To determine how much data Pub/Sub is storing for your topic, you can use Pub/Sub monitoring tools such as Cloud Monitoring. To track the performance and usage of your subscriptions and topics Pub/Sub exposes various metrics that you can monitor. Metrics include the message backlog size, the number of unacknowledged messages, and the message throughput rate. By these metrics we can gain insights into the amount of data stored in your topic and the overall health of your Pub/Sub system.